

## Layout

Layout .....	1
1. Introduction .....	3
1.1. What is SimFlex™ SCL Checker.....	3
1.2. The SimFlex™ tool platform.....	3
1.3. SimFlex™ Client Simulator .....	3
1.4. SimFlex™ SCL Checker .....	3
1.5. What is the SimFlex™ SCL Checker used for? .....	4
How should I use the <i>SimFlex™ SCL Checker</i> ? .....	4
1.6. System requirements .....	5
1.7. Basic skill requirements.....	5
IEC 61850 standard .....	5
Technical Issues (Tissues).....	5
UCA International Users Group testing .....	5
2. Running the <i>SimFlex™ SCL Checker</i> the first time. ....	5
2.1. Start the SimFlex™ SCL Checker .....	6
2.2. Configure the network interface.....	7
2.3. Configure details about the DUT .....	8
2.4. Load the data model from the DUT .....	9
2.5. Load the SCL file (*.icd/*.cid) that belongs to the DUT .....	10
2.6. Select and run your first test case .....	11
3. Simflex SCL checker overview .....	12
3.1. The main application window .....	12
3.2. The Configuration Dialog.....	13
3.3. Details in the General tab.....	14
3.4. Details in the Devices tab.....	15
3.5. Details in the Conditionals tab .....	16
3.6. Details in the Mdl test tab .....	17
4. Use case: Configuring a test .....	18
4.1. Configuration for Cnf3 .....	18
4.2. Configuration for Mdl2 and Mdl3 .....	18
5. Use case: Running a test .....	19
5.1. Select and run a test case.....	19
5.2. Opening and examining a log file .....	20
6. Use Case: Request Support.....	20
6.1. Using Support Feature .....	20

6.2.	Troubleshooting Support Feature .....	22
7.	Advanced Topics.....	23
7.1.	Select a source for Mdl test cases.....	23
7.2.	How to use the data model.....	24
	Tree hierarchy.....	24
	How to read the values in the DUT data model (MMSData values).....	24
	MMSData type and length information: .....	25
8.	Frequently Asked Questions (FAQ).....	26
8.1.	How to install the SimFlex™ SCL Checker?.....	26
8.2.	I cannot connect to my device .....	26
9.	List of possible error messages .....	27
9.1.	Conventions used .....	27
9.2.	Messages in Cnf1 .....	28
9.3.	Messages in Cnf2 .....	28
9.4.	Messages in Cnf3 .....	29
9.5.	Messages in Cnf4 .....	29
9.6.	Messages in Cnf5 .....	29
9.7.	Messages in Mdl1 .....	29
9.8.	Messages in Mdl2 .....	30
9.9.	Messages in Mdl3 .....	31
9.10.	Messages in Mdl4 .....	31
9.11.	Messages in Mdl5 .....	31
9.12.	Messages in Mdl6 .....	31
9.13.	Messages in Mdl7 .....	31
9.14.	Messages in Mdl8 .....	31
9.15.	Messages in Mdl9 .....	31
9.16.	Messages in Mdl10 .....	31
9.17.	Messages in Mdl11 .....	32
9.18.	Messages in all Mdl tests .....	32
9.19.	Messages available in all tests (Cnfx and Mdlx) .....	32
10.	Additional Topics.....	34
10.1.	File name conventions .....	34
10.2.	Used abbreviations .....	34
10.3.	Referenced documents .....	34
10.4.	Referenced tools.....	34
10.5.	Version History.....	34

## **1. Introduction**

SimFlex™ is a software tool platform for testing and simulating Intelligent Electronic Devices (IED) that are based on the IEC 61850 standard.

This manual describes how to use the *SimFlex™ SCL Checker*. For the installation of the *SimFlex™ SCL Checker* please read the “Getting Started” manual.

### ***1.1. What is SimFlex™ SCL Checker***

SCL is an XML based language defined in IEC 61850-6. It can be used for:

- 1) describing the capabilities of an IED (\*.icd)
- 2) describing the configuration of an IED (\*.cid)
- 3) describing the configuration of a system (\*.scl)
- 4) describing the system specification (\*.ssd)

The SCL Checker is designed to test .icd and .cid files against the accompanying IED and the standard.

*SCL Checker* is one of the tools from the *SimFlex™* family. The SCL Checker implements all Cnfx and Mdlx test cases defined in the document “*Conformance Test Procedures for Server Devices with IEC 61850-8-1 interface, Revision 2.3*” from the UCA® International Users Group.

### ***1.2. The SimFlex™ tool platform***

*SimFlex™* is a group of tools developed by GridClone. Some tools within the *SimFlex™* family of tools have been developed for testing an IED or system against the IEC 61850 standard.

### ***1.3. SimFlex™ Client Simulator***

*SimFlex™ Client Simulator* is a tool developed by GridClone. The *Client Simulator* is used for conformance testing an IED, during a test also called the DUT (Device Under Test), as specified in the “*Conformance Test Procedures for Server Devices with IEC 61850-8-1 interface, Revision 2.3*” from the UCA® International Users Group. This document is based on the abstract test cases that are defined in the IEC 61850-10.

### ***1.4. SimFlex™ SCL Checker***

*SimFlex™ SCL Checker* is a tool developed by GridClone. The *SCL Checker* is used for conformance testing of an SCL (\*.icd/\*.cid) file as specified in the “*Conformance Test Procedures for Server Devices with IEC 61850-8-1 interface, Revision 2.3*” from the UCA® International Users Group. This document is based on the abstract test cases that are defined in the IEC 61850-10.

The *SCL Checker* is an easy to use tool with advanced capabilities that enables utilities, manufacturers, system integrators and conformance test laboratories to automatically test IEC 61850 enabled devices. The *SimFlex™ SCL Checker* comes with a set of test cases that implement all the IEC 61850 test cases. The test cases can be individually selected and executed. The *SCL Checker* is easy to use and its intuitive user interface enables to quickly

perform tests. The *SimFlex™ SCL Checker* is a companion tool of the *SimFlex™ Client Simulator*, meaning that together with the *Client Simulator* the *SCL Checker* can be used to test an IED and its SCL file on conformity to the IEC 61850 standard.

This makes the *SimFlex™ Client Simulator* an excellent software solution for the electrical power industry.

### 1.5. What is the *SimFlex™ SCL Checker* used for?

The main goal of the *SCL Checker* is to make the conformance testing as automatic as possible and cover all tests specified by the UCA® International Users Group on the device and the SCL (\*.icd/\*.cid) file. Tests defined by the UCA® International Users Group are (IEC 61850 Ed.1):

Configuration file	Data Model
Cnf1	Mdl1
Cnf2	Mdl2
Cnf3	Mdl3
Cnf4	Mdl4
Cnf5	Mdl5
	Mdl6
	Mdl7
	Mdl8
	Mdl9
	Mdl10
	Mdl11

Not all tests are fully automatic; Cnf1 (validating SCL (\*.icd/\*.cid) against the schema) is also done while opening the SCL (\*.icd/\*.cid) file, Cnf3 needs manual intervention of the test engineer to reconfigure the DUT and reloading a changed SCL (\*.icd/\*.cid) file and Mdl5 cannot be automated at all (this test case will be declared 'deprecated' in IEC 61850 Ed.2).

#### How should I use the *SimFlex™ SCL Checker*?

Typically the user of the *SCL Checker* is a test engineer. The *SCL Checker* is used as a tool that will retrieve information from the DUT (Device Under Test) and its accompanying SCL file (typically an \*.icd/\*.cid file). Each test case (Cnf1 to Cnf5, Mdl1 to Mdl11) can be executed separately or in any combination. The results are stored in a log file (per test case) and the test engineer can review the results of the tests by checking both the log file and, if applicable, the capture file with communication between DUT and the *SimFlex™ SCL Checker*.

To fully use all capabilities of the *SimFlex™ SCL Checker* the tools needs an IED and its accompanying SCL file. The IED must be accessible through a LAN connection. The log files and capture files are stored in some storage facility, by default a folder on the local hard disk.

## **1.6. System requirements**

The *SimFlex™ SCL Checker* has the following software and hardware requirements:

- Windows XP SP3, Windows 7 SP1 or Windows 8
- 1GB of RAM or more
- 1 GHz processor or faster
- 250 MB of free hard disk space
- At least one 100 Mbit Network Interface Card

**IMPORTANT:** Install the *SimFlex™ SCL Checker* software using an account with administrator rights. Before installing the *SimFlex™ SCL Checker* software please make sure that you have installed Microsoft's .Net framework version 4.0 Full Profile (or better) on your computer. Without installing .Net Full Profile the *SimFlex™* installer and/or the program itself will not run properly. Windows 8 users do not have to install the .Net Full Profile as it is already part of the Operating System.

The .Net Full Profile installation file can be downloaded from the Microsoft web site:

<http://www.microsoft.com/en-us/download/details.aspx?id=17718>

## **1.7. Basic skill requirements**

The test engineer must have knowledge about the IEC 61850:2003 standard (Edition 1), the tissues (technical issues) related to this standard and the “*Conformance Test Procedures for Server Devices with IEC 61850-8-1 interface, Revision 2.3*” from the UCA® International Users Group.

### **IEC 61850 standard**

The IEC 61850:2003 can be purchased on the IEC web site. See: <http://webstore.iec.ch>

### **Technical Issues (Tissues)**

Information about technical issues is available at: <http://www.tissues.iec61850.com>

### **UCA International Users Group testing.**

Information about the testing procedures is available on the UCA® International Users Group website. See: <http://www.ucaiug.org>

Please be aware that to access certain documents on the UCA® web site, you might need to become a member of the UCA® International Users Group.

## **2. Running the *SimFlex™ SCL Checker* the first time.**

Running your first test with the *SCL Checker* involves some configuration steps.

We express the importance of following the steps as described below even when you are eager to start with the SCL checker. *Not following this procedure will result in problems and a*

*non-functional installation.* The installation procedure is described step by step with screen shots.

Some of these steps are needed only once. The information from it is stored in the program's configuration database.

Most of the information in the subsections is also available as a separate section with a more detailed explanation.

Steps involved:

- Start your copy of the *SimFlex™ SCL Checker*
- Configure the network interface
- Configure details about the DUT (Device Under Test)
- Load the data model from the DUT
- Load SCL (\*.icd/\*.cid) file that belongs to the DUT
- Select and run your first test

## 2.1. Start the *SimFlex™ SCL Checker*

Locate the *SimFlex™ SCL Checker* icon on your desk top or in your Start menu and (double) click on it. The main screen of the program will open.



The Desktop Icon on top, right, the menu icon left of it and the main screen below.



Figure 1 *SimFlex™ SCL Checker* links

Enter or change the name of test engineer that will use the program for testing

A dialog box for the log on procedure. It has a light gray background and a blue border. On the left, the label 'UserName' is followed by a text input field containing the placeholder text 'Your name should be here'. On the right side, there are two buttons: 'Done' with a green checkmark icon and 'Cancel' with a red 'X' icon.

Figure 2 Log on procedure

Press [ **Done** ] when ready.

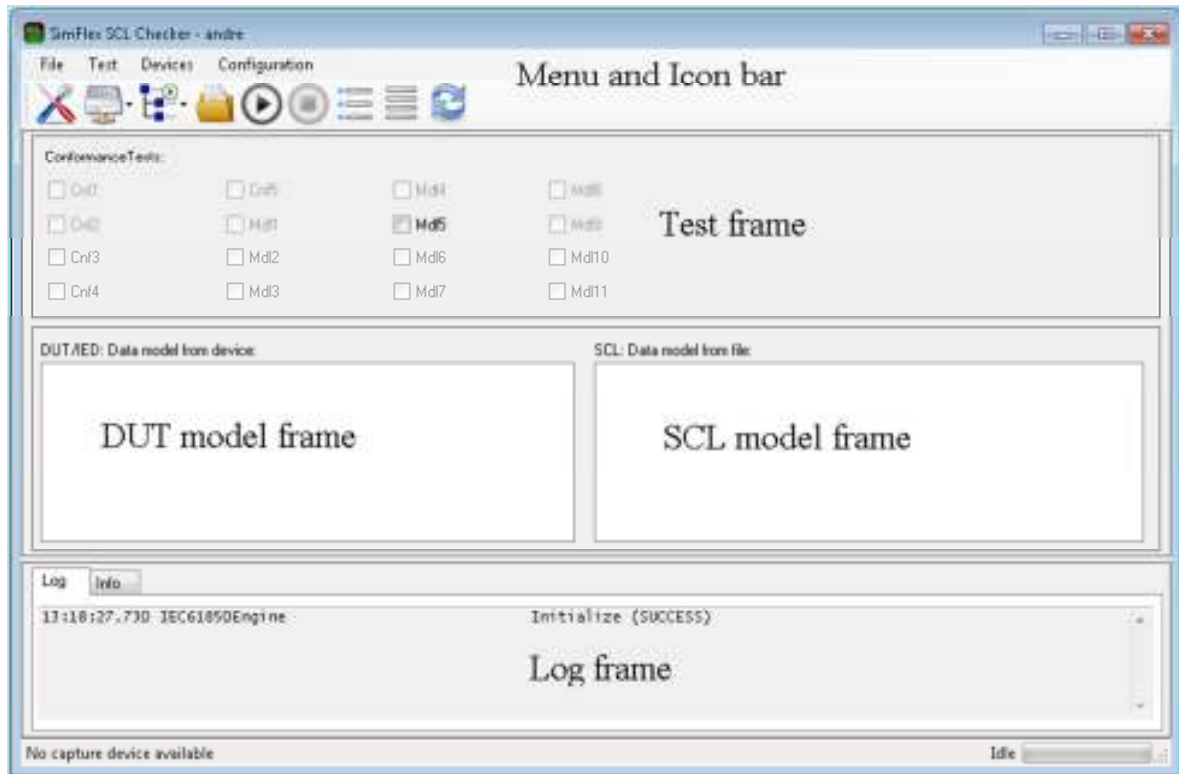


Figure 3 *SimFlex™ SCL Checker* Main screen

## 2.2. Configure the network interface

The next step is to configure the *SimFlex™ SCL Checker* to use the correct IP information for connecting to the DUT.



Click on the Configuration Icon (shown left) in the icon bar and select the [ **General** ] tab in the opening form (shown below).

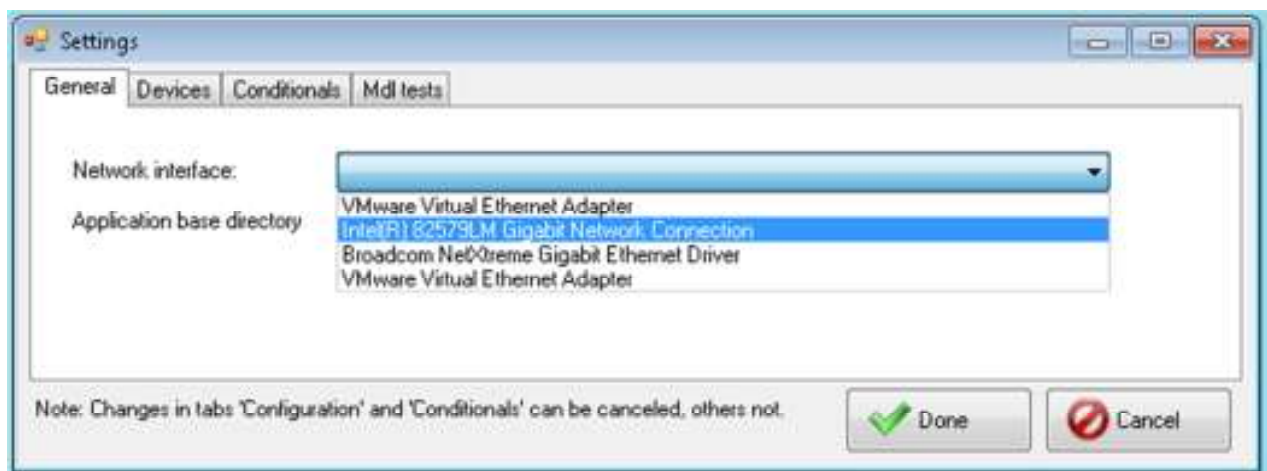


Figure 4: Selecting the NIC (Network Interface Card)

Select the network interface that you want to use for communication with the DUT.

Press the [ **Done** ] button when ready.

### 2.3. *Configure details about the DUT*

This needs to be done only once per model. Use the **Devices** → **Add Device** menu entry to open the device form. You can also use the [ **Device** ] → **Add Device** icon for that.

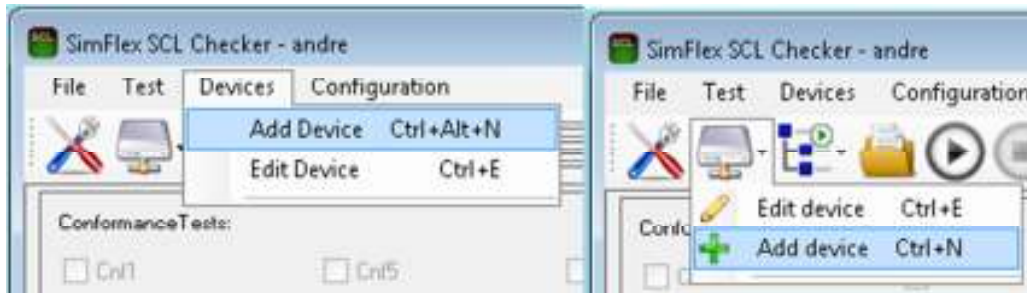


Figure 5: Open the Device Configuration dialog

The device form shows all the details, only a few are needed for a Quick start.



Figure 6: Device configuration.

Fill in a device name and fill in the known IP number for the device. Click [ **Done** ] when ready.

After all details are filled in you can select your device via the [ **Devices** ] menu or the [ **Device** ] icon.

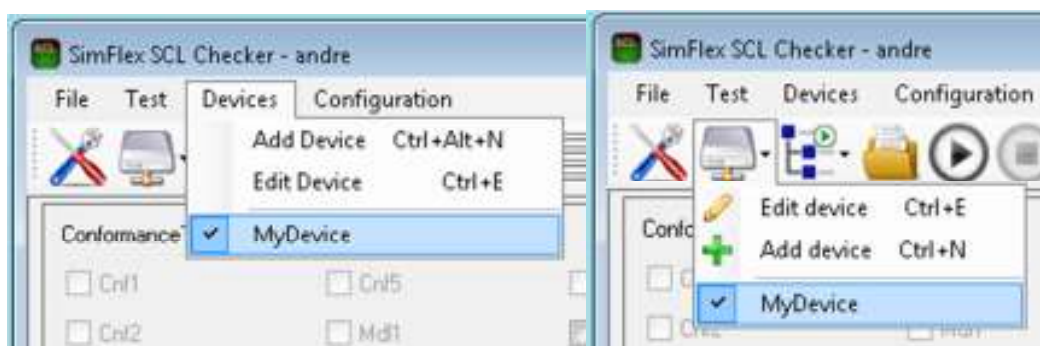


Figure 7: Select a device



You can see which device is selected on the left side of the screen, in the DUT frame.



Figure 8: The DUT frame shows which device is selected

## 2.4. Load the data model from the DUT

Once you have configured and selected your device in *SimFlex™ SCL Checker* you can connect to it and let the program build a data model for it. This data model will become available to you in the data model frame. While building the data model the log frame at the bottom will inform you about the progress.

Use the [ **Data model** ] → **Build data model** icon in the tool bar to retrieve the data model from the DUT. There is no menu item to start this.

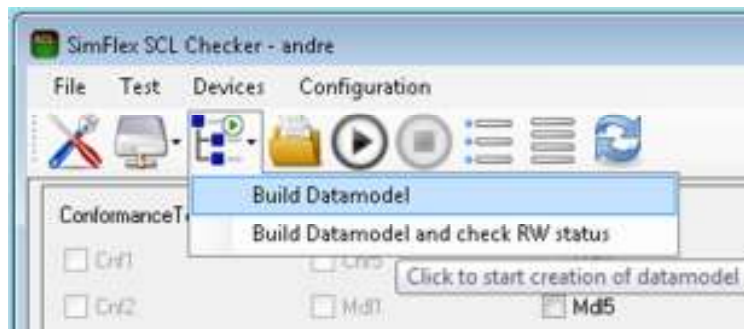


Figure 9: Build a data model

You can right-click on the device name in the DUT frame and select [ **Build data model** ] in the popup window.

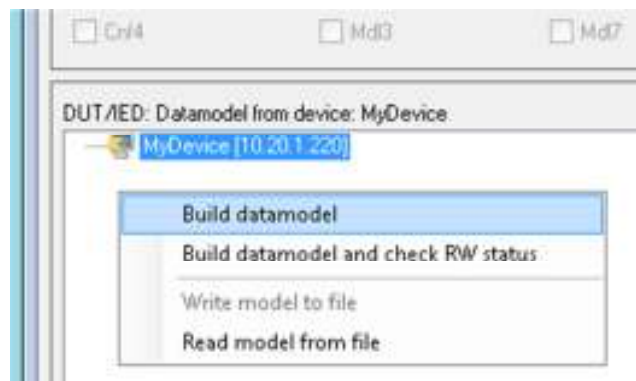


Figure 10: Build a data model

After building the model you should see a model icon in the data model frame, indicating that information about the data model is available.



Figure 11: Data model retrieved from the DUT

## 2.5. Load the SCL file (\*.icd/\*.cid) that belongs to the DUT

Most of the validation by the *SimFlex™ SCL Checker* is not done with the DUT but with the SCL file describing the DUT, its capabilities and configuration. This file must be loaded as well. Use the [ **Open configuration file** ] icon or right-click on the SCL model frame and select [ **Build data model** ] to start loading it.

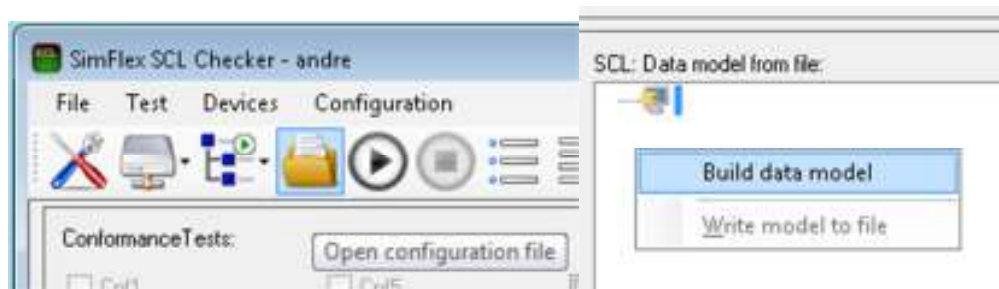


Figure 12: Open SCL file of the DUT.

A file selector window will open to enable you to select the correct SCL file (\*.icd/\*.cid).

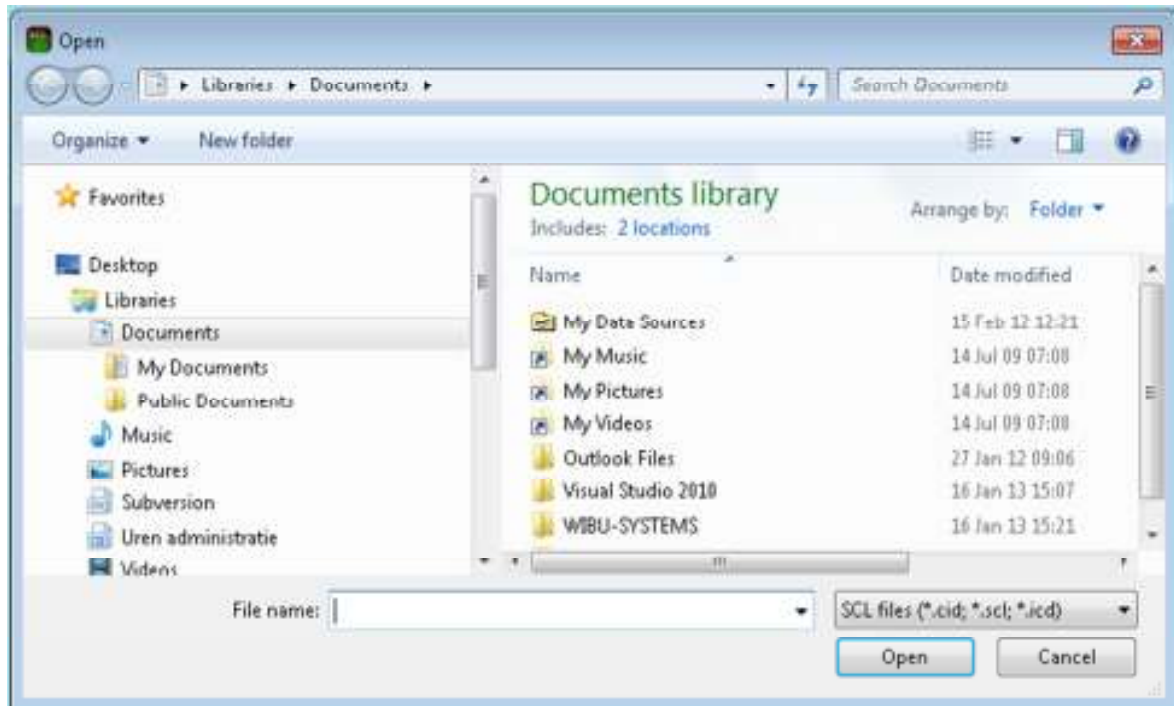


Figure 13 File selector window

Select the correct SCL file and click [ **Open** ].

When the file is loaded a data model will be built from it and will be shown in the SCL model frame.

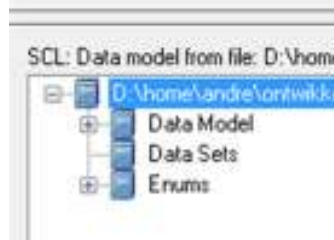


Figure 14 SCL data model

## 2.6. *Select and run your first test case*



Figure 15: A selected test case

When both DUT model and SCL model are loaded all test cases are available. Select Cnf2 as the first one to run (Cnf1 is a 'formal' test, already failing when the SCL file cannot be loaded.) You are now ready to run the Cnf2 test. Do so by clicking on the [ **Run** ] icon (See picture below, left side.)



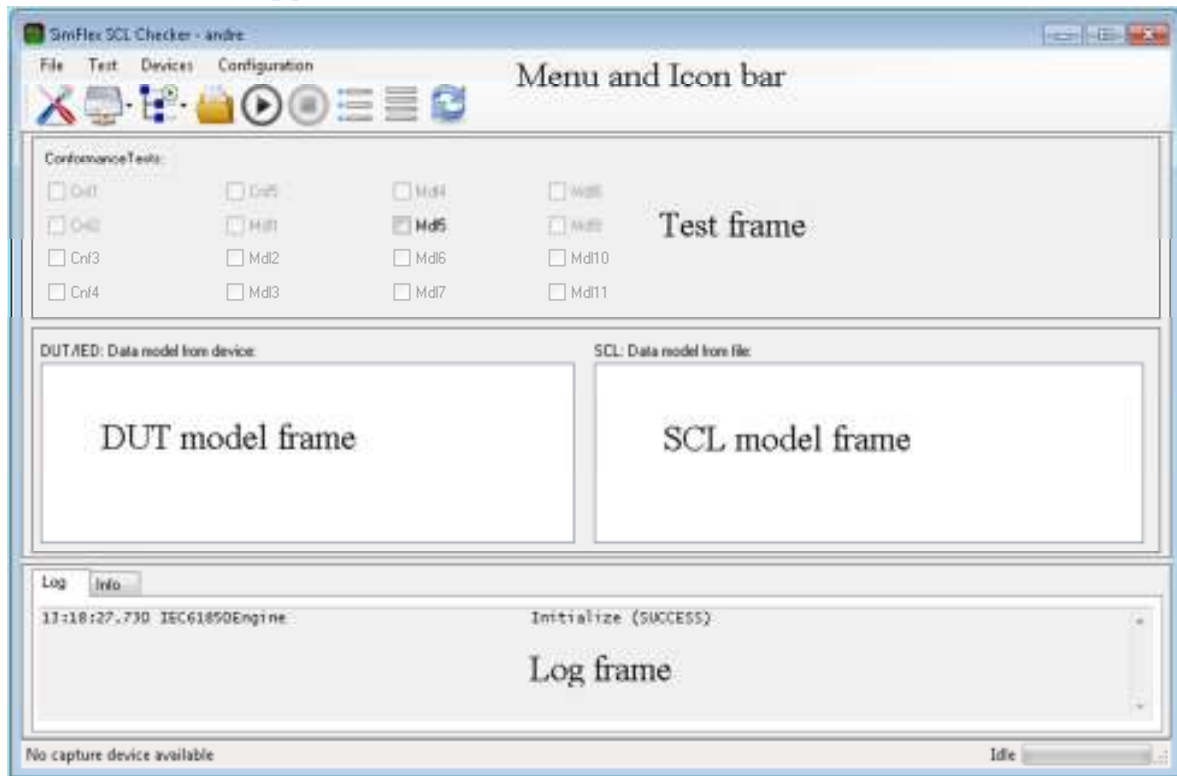
Figure 16: Start and stop a test

Push the button to start the test. The [ **Run** ] icon is disabled during the test.







### 3. Simflex SCL checker overview





This section describes the components that are visible in the main screen of the *SimFlex™ SCL Checker*.

#### 3.1. The main application window



The main components of the application window:

- A menu and Icon bar with (from left to right)
  -  The configuration icon
  -  The device icon/submenu
  -  The DUT model icon/submenu.
  -  The SCL (\*.icd/\*.cid) model icon.
  -  The Run selected test icon.
  -  The abort test icon.

-  The select all tests icon.
  -  The deselect all tests icon.
  -  The deselect and reset all tests icon.
  -  The Support icon
- The test case frame.  
This frame shows all tests cases (Cnf1 to Cnf5, Mdl1 to Mdl5).  
Available tests are enabled, not available tests are disabled.  
A test becomes available when the data model for it (DUT and/or SCL) has been retrieved from the DUT or the applicable SCL file has been loaded..
  - The DUT model frame.  
Load the data model from the DUT by right-clicking on the frame and then select “Build data model” to build a data model with Readable information only. Select “Build data model and check RW status” for full Readable/Writable information.  
You can also use the DUT model icon/submenu for this.
  - The SCL (\*.icd/\*.cid) model frame.  
Load the data model from the SCL file by right-clicking on the frame and then select “Build data model”, select the proper SCL (\*.icd/\*.cid) file in the upcoming file selector window.
  - The Log and info frame.  
This frame will keep all messages, warnings and other information and communication status messages.

### ***3.2. The Configuration Dialog***

The Configuration Dialog shows 4 tabs. Each tab contains items that can be configured. In the next parts each tab and its configuration items are displayed and described.

### 3.3. Details in the General tab

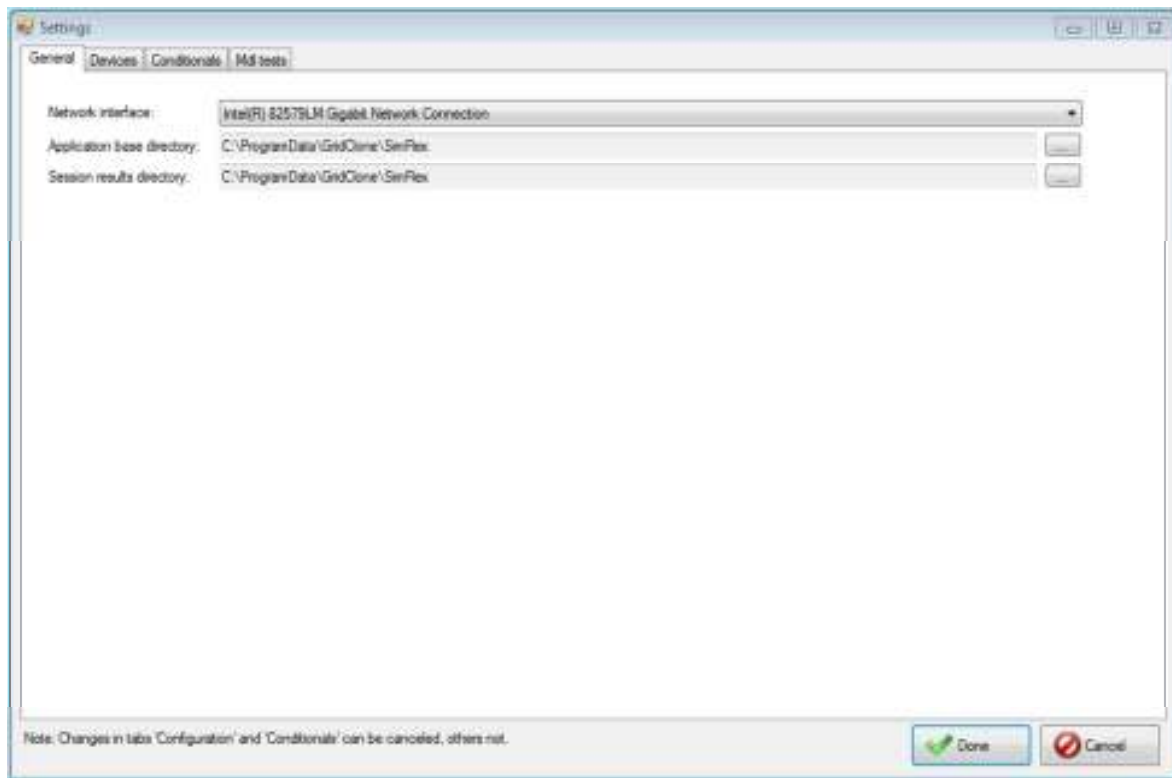


Figure 17 [ General ] tab for selecting the NIC (Network Interface Card) to use for communication with the DUT

Available items are:

- The Network Interface dropdown list.
- The “Application base directory”.
- The “Session Results directory”.

Note: This configuration item is shared with the *SimFlex™ Client Simulator*. Changing it here will also change it for the *Client Simulator*.

### 3.4. Details in the Devices tab

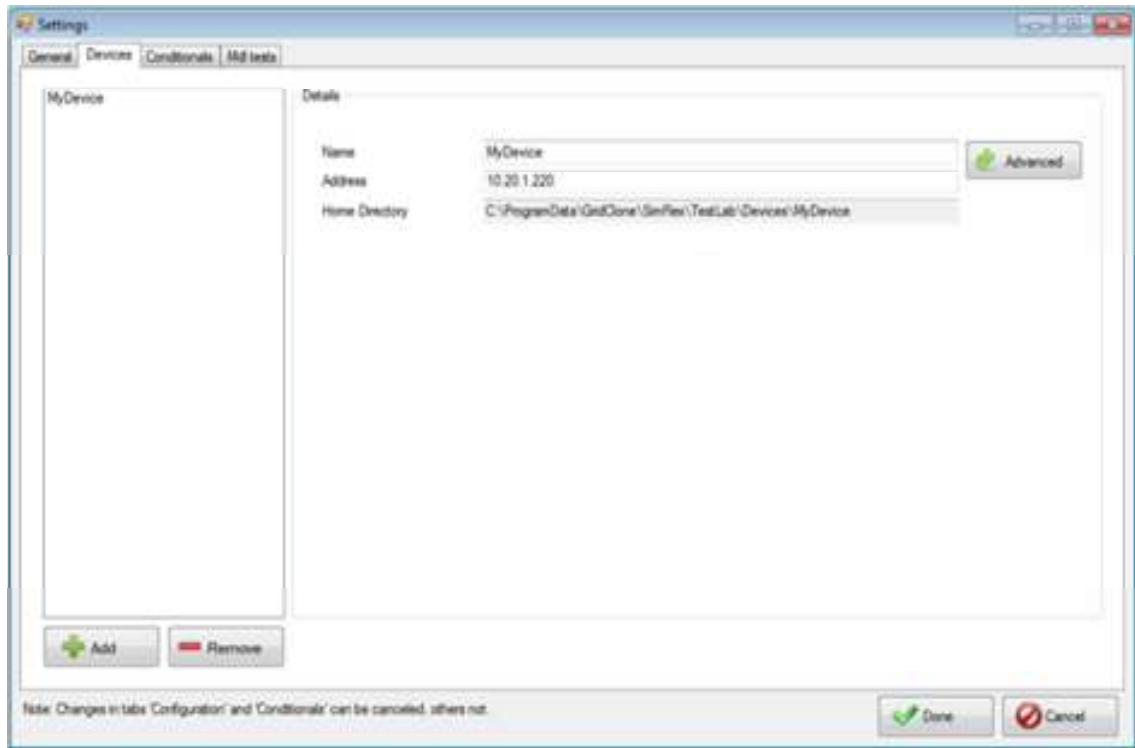


Figure 18 [ Devices ] tab for managing data for several DUTs

Use the [ **Remove** ] button to remove the selected device.

Use the [ **Add** ] button to add a new device/DUT.

Edit the [ **Name** ] field to reflect the name of your device/DUT.

Edit the [ **Address** ] field to reflect the IP address of your device/DUT.

Use the [ **Data model** ] field to bind a fixed SCL file to your device/DUT. When an SCL file is bound to the device/DUT or is loaded via the main menu the *SCL Checker* will always try to load this file when the device/DUT is selected. This will result in communication messages in the log window.

Note: All settings in this tab are shared with the *SimFlex™ Client Simulator* tool.

### 3.5. Details in the Conditionals tab

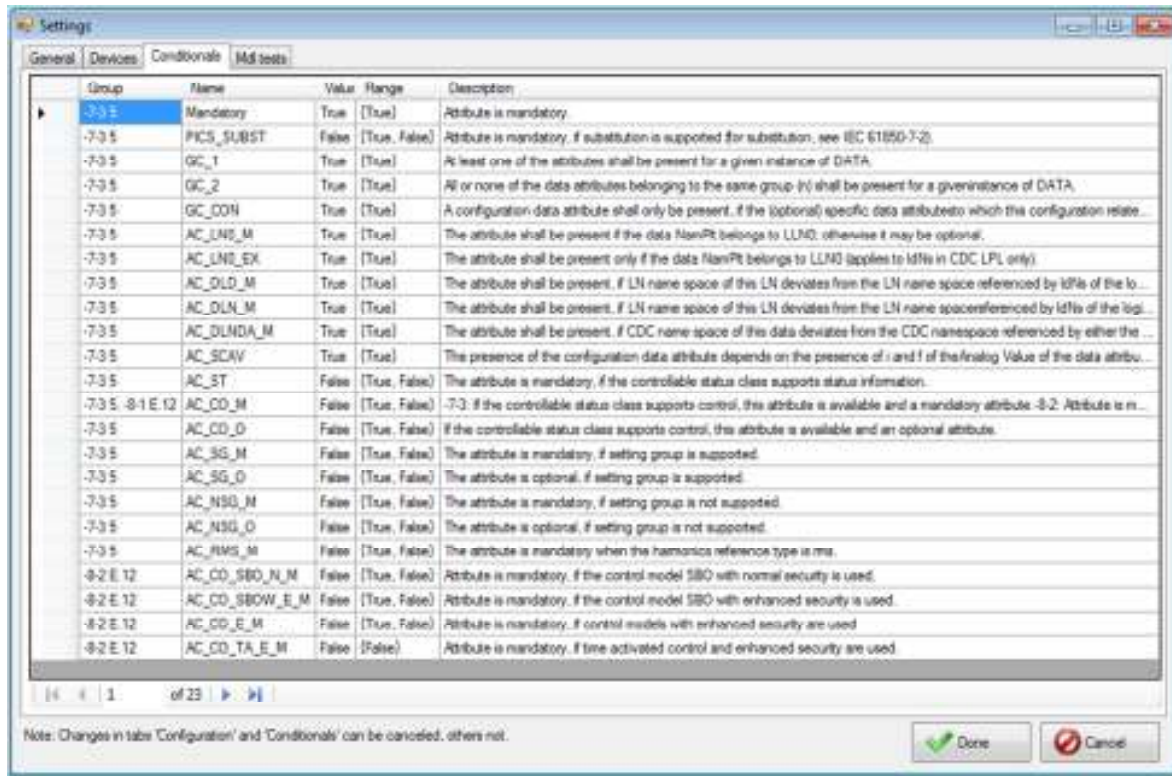


Figure 19 [ Conditionals ] tab for configuring PIS/PIXIT information

Description of the columns:

- Group** Reference to the IEC 61850 part and chapter where this condition is specified. Fixed, for reference only.
- Name** Name of the condition (as described in IEC 61850). Fixed, for reference only.
- Value** True (the condition is applicable) or False (the condition is **not** applicable). Can be set to a value available in the Range column (next column to the right.)
- Range** Accepted values for the previous column. If only one value is available in this column then the Value column is set and fixed to this value. This column is informative. Fixed, for reference only.
- Description** A description of the condition as given in the IEC 61850. This column is informative. Fixed, for reference only.



### 3.6. Details in the Mdl test tab

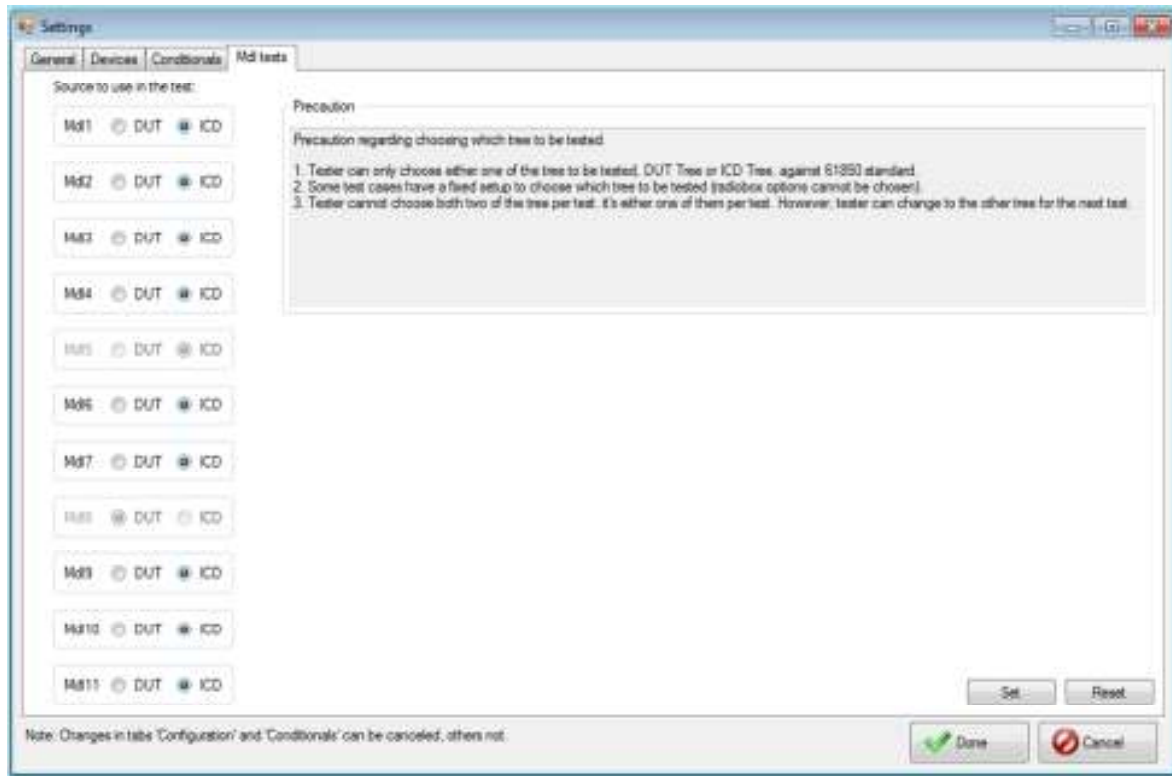


Figure 20 [ Mdl tests ] tab for switching test cases between the DUT model and SCL model for some Mdl tests

In this configuration screen the test engineer can select whether a test cases is executed based on the DUT data model or the SCL data model. In some cases the SCL file of a DUT is not available, but the test engineer still wants to verify if the DUT complies with SCL rules.

The [ **Set** ] button will fix the source of the specific test case for the duration of the program. Once the *SCL Checker* program is closed all settings in the *Mdl test* dialog are restored to their original values.

The [ **Reset** ] button will restore all settings in the *Mdl test* dialog to their original values.

Note: In the *SCL Checker* the test cases Cnf1-Cnf5 and Mdl1-Mdl11 are available based on the data models retrieved from the DUT or loaded from the SCL file. When in the *Mdl test* dialog a source (data model) is selected that is not available in the program then the applicable test will result in an error when executed.

Example:

Mdl1 is using the SCL file as its default source. However the test engineer has only retrieved the data model from the DUT and has not loaded the SCL file, yet. With the setting for Mdl1 as shown in Figure 20 the execution of test case Mdl1 will result in an error message:

Mdl1: Start: Mdl1

Mdl1: Test description: Verify presence of mandatory objects for each LN. ...

Mdl1: *No valid SCL data from file available (test inconclusive)*

Mdl1: Done: Mdl1

## **4. Use case: Configuring a test**

Configuration for tests is only needed for Cnf3, Mdl2 and Mdl3. See the next chapters on how to configure these test cases.

### ***4.1. Configuration for Cnf3***

This test needs an additional SCL (\*.icd/\*.cid) file with at least five relevant changes in it. (See the UCA test description for more details).

The DUT must be configured during this test with the changed SCL (\*.icd/\*.cid) file and the *SimFlex™ SCL Checker* will ask the test engineer during Cnf3 to select the SCL (\*.icd/\*.cid) file and to reconfigure the DUT.

Please refer to the manual of the DUT how to update its configuration.

### ***4.2. Configuration for Mdl2 and Mdl3***

Mdl2 and Mdl3 check several (21) different conditions. These conditions are partly fixed and partly depending on the manufacturer. The manufacturer must supply this information through the PICS/PIXIT documents. This information must be included in the configuration's **[ Conditionals ]** tab.

Steps to be taken for configuring the PICS/PIXIT information:

1. Open the configuration form and select the **[ Conditionals ]** tab by selecting the configuration icon or selecting the menu **Configuration → Conditionals (PIXIT...)**.
2. For each line in the list, check the setting in the "Value" column.  
The "Group" column is a reference to the IEC61850 document and chapter that is applicable.  
The "Name" column is the name for the condition as used in the IEC61850 documents.  
The "Value" column is the current value (and is the only column that can be edited).  
The "Range" column sums up all acceptable values, if only one value is given then the "Value" is 'fixed' to this single value and cannot be changed.  
The "Description" column is the description for the condition as given in the IEC61850.

The configuration data for Mdl2 and Mdl3 is stored in the "Session results directory".

## 5. Use case: Running a test

### 5.1. *Select and run a test case*



Figure 21: A selected test case

When both DUT model and SCL model are loaded all tests are enabled. Select Cnf2 as the first one to run (Cnf1 is a 'formal' test, already failing when the SCL file cannot be loaded.) You are now ready to run the Cnf2 test. Do so by clicking on the **[ Run ]** icon (See picture below, left side.)



Figure 22: Start and stop a test.

Push the button and the test starts. Some things to notice are:

- The **[ Run ]** icon is now disabled.
- A lot of activities in the log frame at the bottom.

After a normal run the **[ Run ]** icon is enabled again and the test is marked as 'Passed' (green tag), 'Inconclusive' (blue question mark) or 'Failed' (red cross).

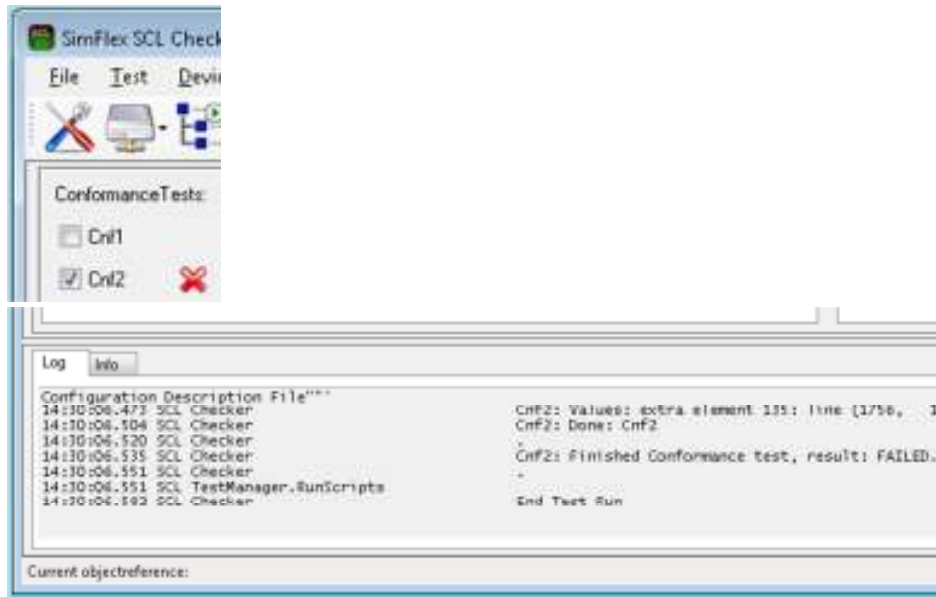


Figure 23: Negative test result

Inspect the log for the exact reason(s) of failing.

## 5.2. Opening and examining a log file

The default area for storing log and capture files is “C:\ProgramData\GridClone\SimFlex”. It is possible to use any location. To configure this see the configuration, [ **General** ] tab, ‘Session results directory’. Select the [ ... ] button at the right side of the setting to change the location.

At this moment in time it is not possible to open the files via the *SimFlex™ SCL Checker*. Open the location in your favourite file manager, select the file and open it with your favourite text editor.

The *SimFlex™ SCL Checker* does not have the same session features as used in the *SimFlex™ Client Simulator*. However, to store test results produced by the *SCL Checker* in the same directory as a *Client Simulator*’s session, the test engineer can specify the directory in the [ **General** ] tab of the configuration dialog. The directory can be specified by browsing for the directory with the [...] button at the right-hand side of the field “Session results directory”. The selected directory is displayed in the lower left corner of the main window.

## 6. Use Case: Request Support

If you’re having trouble using SimFlex SCL Checker, you can send email to GridClone. To understand the condition of your problem, GridClone needs to know the logs and configurations of your SimFlex SCL Checker application.

### 6.1. Using Support Feature

You can use Support feature from the application to send email for requesting support. This feature can be run by click the Support button in the icon bar. This button is available after DUT model or SCL model are loaded. While the test is running, this button will be disabled.

Follow this step to use support feature.

1. Click the Support button from the icon bar

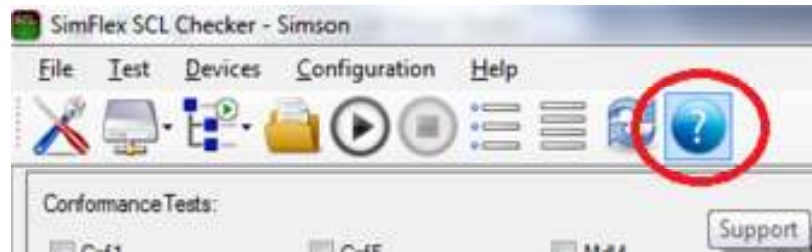


Figure 24 Support Icon

2. A message box will appear. Click OK button.

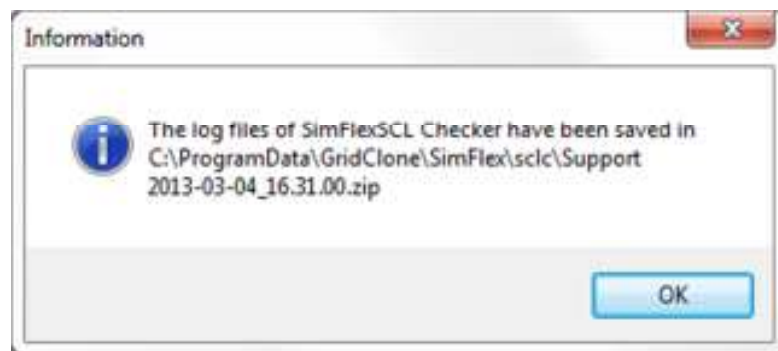


Figure 25 Message Box that shows the location of the Log files

This dialog window gives you information about the location of the log files have been saved.

3. A new dialog will appear. Click Yes button if you want send the support email.

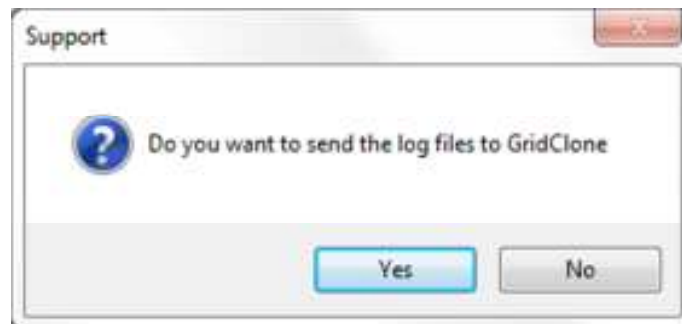


Figure 26 Dialog Window about Sending Support Email

4. Finish your email and send it to GridClone

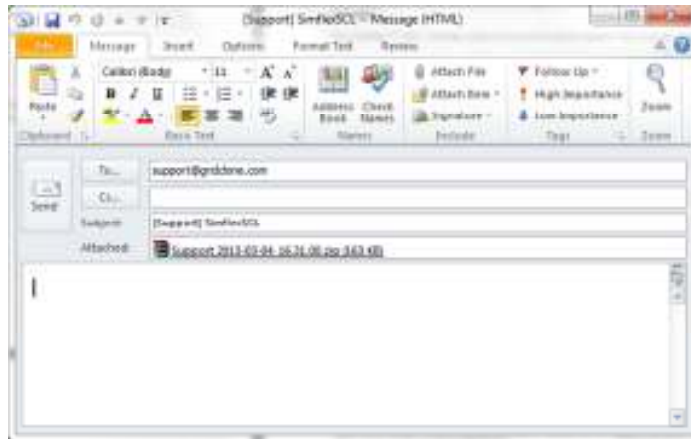


Figure 27 Composing Support Email

### 6.2. Troubleshooting Support Feature

This feature works if you have default mail application on your computer. However, if the email pop-up doesn't appear, you still can manually send support email to GridClone including the log files.

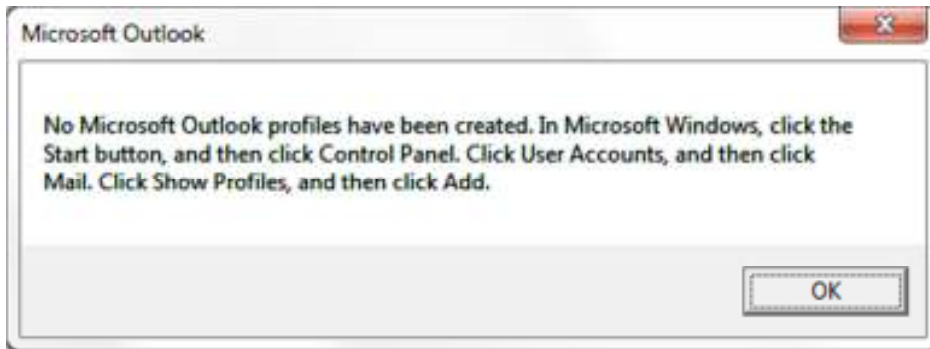


Figure 28 Example of Sending Support Problem

The log files are saved in a zip file. The location of this zip file is written in the previous message box and you can also find it in log frame.

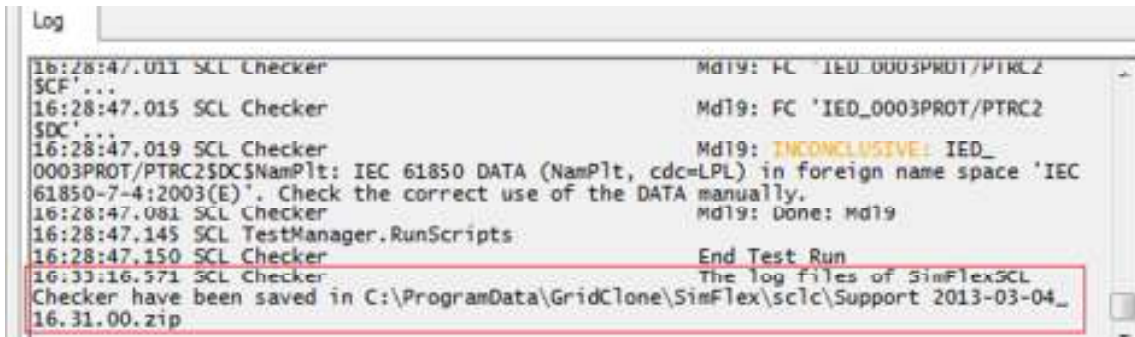


Figure 29 Example of the log message that mention the location of log files

## 7. Advanced Topics

This section includes more information on test cases and data models.

### 7.1. Select a source for Mdl test cases

It is possible to select 'the other model' as a source for *Mdl* test cases. Most tests will work on the DUT model just as well as on the SCL (\*icd/\*cid) file. It is possible to change the source to use from SCL file to DUT model or the other way around, but only for the duration that the program is active. It is reset to default values as soon as the program is terminated.

To change the source open the configuration form and select the *Mdl test* tab.

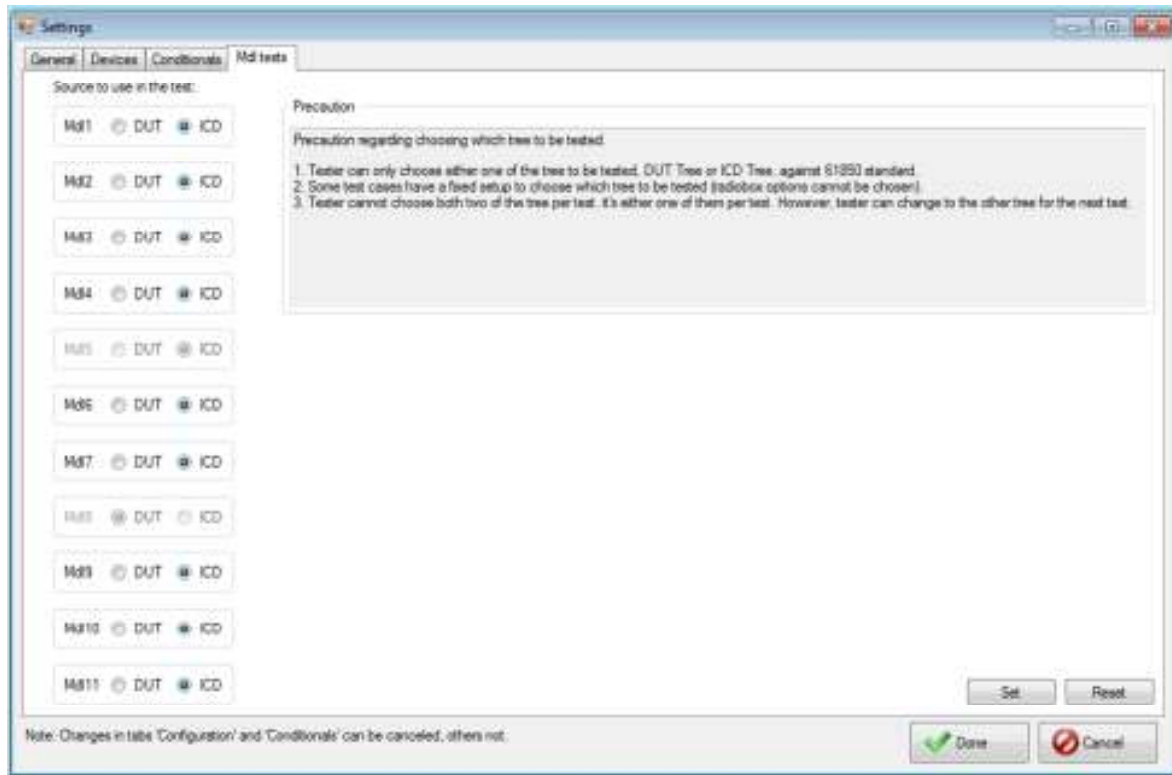


Figure 30: Mdl test data sources

Select the source you want to use for the current test. Any changes are marked in red but not yet active. Press the [ **Set** ] key to activate all your changes.

Use the [ **Reset** ] button to reset the default values.

Select [ **Done** ] when ready.

Note 1:

- Mdl5 does not use any source.
- Mdl8 need both sources.

Note 2:

Tests are enabled if one of the two sources is loaded but will result in an error when the other, not loaded, source is referenced.

Example:

Mdl1 is using the SCL (\*.icd/\*.cid) file as its default source. The DUT data model is also acceptable. In the next run the DUT model is loaded but the SCL file not and the source to use is the SCL file and not the DUT model. This will result in an error message. See below.

Mdl1: Start: Mdl1

Mdl1: Test description: Verify presence of mandatory objects for each LN. ...

Mdl1: *No valid SCL data from file available (test inconclusive)*

Mdl1: Done: Mdl1

## 7.2. *How to use the data model*

### **Tree hierarchy**

LD = Logical Device

LN = Logical Node

FC = Functional Constraint

DO = Data Object

DA = Data Attributes

### **How to read the values in the DUT data model (MMSData values)**

All values from and to a DUT are communicated using the MMS protocol (ISO 9506), MMS for short, as described in IEC 61850-6. Values shown in the DUT data model are restricted to the format described within the MMS protocol.

The values displayed in the DUT data model do not only contain the MMSData type but also includes some length information, a flag indicating if the value is 'Readable', 'Writable' or both and the actual value retrieved from the DUT.

The normal format for an item with a value to is in the data model is:

**Name, [MMSData type], Readable/Writable flag, value.**

Example:

DUT\_NAME [DUT IP address]

Data Model

LD name

FC ST



DO Mod

DA stVal, [INTEGER~8], R., value = 1

DA q, [BITSTRING~-13], R., value= [0000000000000]

DA t, [UTC\_TIME], R., value = u23-11-2012\_17:05:57.720

**MMSData type and length information:**

The length information with the MMSData type is according to IEC 61850, paragraph 8.1.1, table 14, mapping of ACSI data types.

Most MMSData types with a fixed length (such as UTC\_TIME) have no length information, with the exception of BOOLEAN. INTEGER and UNSIGNED which have length information of one digit (8 for 8 bits, 16 for 16 bits etc.). String based types (such as VISIBLE\_STRING) have length information that can be negative indicating a Maximum length (the string may be shorter than the length but only if the length information is negative.). And FLOATINGPOINT values have two length information values, one to indicate the total length (in bits) and the second to indicate the length of the exponent (in bits).

**Readable/Writable:**

At maximum two characters indicate the data attribute readable/writable information. The possible characters are:

Character	Meaning
-	The Readable/Writable state is unknown.
.	The Readable/Writable state is untested.
R	The data attribute is Readable
W	The data attribute is Writable
w	The data attribute should be writable but is not available at the moment for writing to.

The most common combinations are:

Combination	Meaning
-	The data attribute Readable/Writable state is unknown.
.	The data attribute Readable/Writable state is untested.
R.	The data attribute is Readable but not tested for Writable.
R-	The data attribute is Readable but not Writable.

- Rw            The data attribute is Readable and should be Writable but is not available for writing at the moment.
- RW            The data attribute is Readable and Writable.

## **8. Frequently Asked Questions (FAQ)**

Some questions do reoccur and become frequently asked (that is, more than once). Check this section before you contact GridClone for support on your question.

### ***8.1. How to install the SimFlex™ SCL Checker?***

**Question:**

How do I install the *SimFlex™ SCL Checker* program?

**Answer:**

Read the *Getting started* document that comes with the program for a detailed description of the installation procedure. If you do not have **administrator rights** on your PC, please contact your IT department.

### ***8.2. I cannot connect to my device***

**Question:**

How can I check if my device is available on the network?

**Answer:**

See that your DUT is turned on and connected to the same LAN as your *SCL Checker*. Find out the IP address of the DUT.

A simple test to check if the DUT is connected and within range of the *SCL Checker* is to open a Windows Command Window (cmd.exe) and typing the following instruction at the caret:

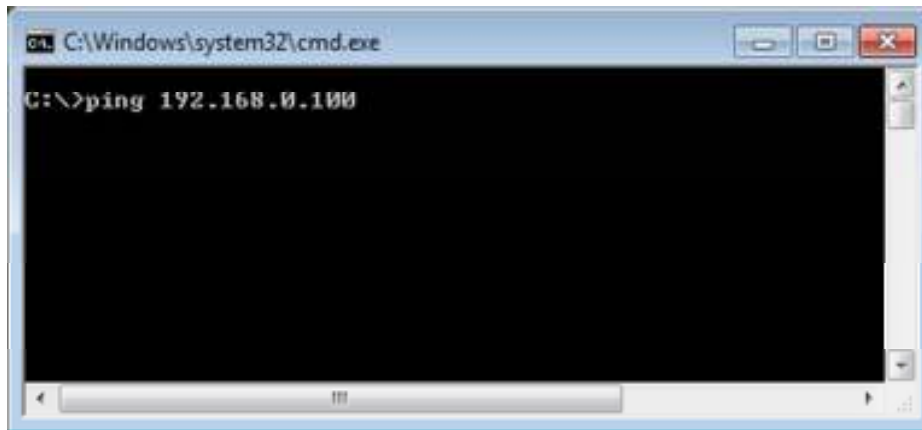


Figure 31: Check DUT network connection.

The IP address behind the *Ping* command shall be the IP address of the DUT. In case the DUT is visible it will react with positive responds.

Next, be sure that you selected the appropriate **network interface** in the [ **General** ] tab in the Configuration Dialog of the *SCL Checker*.

After that be sure that you created a DUT in the [ **Devices** ] tab in the Configuration Dialog of the *SCL Checker*. Check if the IP address is the same as the IP address of your DUT.

Try to retrieve the data model of the DUT.

## 9. List of possible error messages

The *SimFlex™ SCL Checker* has three different kinds of error messages.

- 1) Error messages resulting from a test case
- 2) General error messages
- 3) BUG messages. (Please report these messages to GridClone).

*SimFlex SCL Checker* has two levels of verbosity to report messages, “Normal mode” and “Verbose mode”. You can switch between the two by (de)selecting the menu **Test → Verbose** option.

In “Normal mode” (**Test → Verbose** unselected), only messages that will cause a test to fail or to be inconclusive is reported.

In “Verbose mode” (**Test → Verbose** selected) any decision made in a test is reported. This enables a test engineer to precisely follow all actions by the test and to verify not only why a test is failing but also that a test is passing for the right reasons.

**WARNING:** switching on the **Test → Verbose** flag will generate huge amounts of logging data and can extend the time it takes to finalize a test.

To find more information about an error message please look up the messages for a particular test and if no information is found, try the general section. These sections give details on the “Normal mode” messages. The extra “Verbose mode” messages follow a similar pattern and should be clear from the context and content alone. If not, please get in contact with GridClone.

Bug messages are clearly identified with the first (or last) word “BUG: ...”. In normal circumstances this kind of message will not show up but in case the program or parts of its data gets corrupted and the internal reflections and checking done by the program comes up with an unexpected, unexplainable or impossible result this message is given. Any details on how the message was triggered, what SCL files and DUT are used will be helpful to solve the problem.

### 9.1. Conventions used

Any message that will cause a test to fail has the word “ERROR:” at the beginning of the line.

E.g: “ERROR: line (1846, 11): DA MyLogicalDevice/LLN0\$CO\$Mod\$Oper: Missing ‘Check’”

*<Line number>* A position in a SCL file given as (line, column) combination.

E.g. “line (1846, 11):”

A *<Line number>* value is normally followed by an *<Object Reference>* value. A *<line number>* is given when available and left out (including the colon) when not available. A *<line number>* refers to the line and column in the SCL file where a data object or value is specified. When data objects between the DUT model and SCL (\*.icd/\*.cid) file are matched then any message based on data objects from the DUT model may include *<line number>* information on the matched data object in the SCL (\*.icd/\*.cid) file. (Matching data objects between DUT and SCL (\*.icd) file is done on Data Object Reference only.)

*<bType>* Part of Basic IEC61850 Type information. bType is one of Int8, Int16, Enum, etc

Note: Type information in IEC61850 is defined as a (bType, type) tuple where the type part can be empty.

*<Object class>* Data object classification, e.g. “LD”, “LN”, “FC”, “DO” etc.

*<Object name>* Data object name, e.g. “LLN0”, “Mod”, “Oper”, “Check” etc.

*<Object Reference>* A value of an object reference e.g.  
“MyLogicalDevice/LLN0\$CO\$Mod\$Oper\$Check”

An *<Object Reference>* is usually preceded with a *<Line number>* value.

*<Result>* Any test result, one out of “PASSED”, “INCONCLUSIVE” or “FAILED”

*<Test>* Test name, e.g. “Ctl1”, “Ctl2”, ... “Mdl1”, “Mdl2”,...

*<type>* Part of Basic IEC61850 Type information. ‘type’ is a further specification on the bType part, most commonly given with bType is Struct, Array or Enum

Note: Type information in IEC61850 is defined as a (bType, type) tuple where the type part can be empty.

## 9.2. Messages in Cnf1

Cnf1 does not generate test specific error messages. The SCL (\*.icd/\*.cid) file cannot be loaded in cases where this particular test would fail and as such you cannot start this test. A catch 21 situation or in another view on it: Test Ctl1 is done while loading the file.

If you cannot load a SCL (\*.icd/\*.cid) file, check the log window at that moment, for particular details why the file cannot be loaded. The log must include at least one line with particular information (“Line (x, y): reason of failing”) to indicate the problem where x is the line number and y is the column number in the file where the problem is located.

The actual test for Cnf1 is redone but cannot fail.

## 9.3. Messages in Cnf2

Message: “Missing *<count>* objects in SCL (\*.icd/\*.cid) that are referenced in data set(s).”

Meaning: A number of objects that must be available according to the SCL (\*.icd/\*.cid) data sets are not found in the file. This line is followed by a list of object references missing.

Message: “Missing <count> objects in DUT that are referenced in data set(s).”

Meaning: A number of objects that must be available according to the SCL (\*.icd/\*.cid) data sets are not found in the file. This line is followed by a list of object references missing.

Message: “ERROR: <test part>: Found <count> element(s) in SCL (\*.icd/\*.cid) file which are not available in DUT model.”

Meaning: Cnf2 has four parts, checking names, types, data sets and pre-set values. In each of these parts items can be found missing. A list of object references of the missing parts is given.

Message: “ERROR: <test part>: Found <count> element(s) in SCL (\*.icd/\*.cid) file which are not available in DUT model.”

Meaning: Cnf2 has four parts, checking names, types, data sets and pre-set values. In each of these parts items can be found missing.

#### **9.4. Messages in Cnf3**

This test has the same messages as Cnf2. See the messages at Cnf2.

#### **9.5. Messages in Cnf4**

Message: “ERROR: Missing <count> MMS optional service(s).”

Meaning: A number of services as specified in the SCL (\*.icd/\*.cid) file are not available in the DUT. (A list of missing services is already given.)

#### **9.6. Messages in Cnf5**

Message: “ERROR: <number>: <object reference>.”

Meaning: A ctlModel values in the DUT is different than specified in the SCL (\*.icd/\*.cid) file.

#### **9.7. Messages in Mdl1**

Message: “ERROR: LNodeType <Object Name>: LD <Object Reference>: Mandatory LN not available.”

Meaning: Some mandatory data attribute is missing in a DA. Both the name of the data attribute and the object reference of the DA are given.

Message: “LNodeType <Object Name>: DO <Object Reference>: LN <object reference>, inconclusive, please check manually”

Meaning: Some DO (with given name) is unknown, possibly due to being in another name space. Please check if this LN is in another namespace and confirm the correctness of this DO not being available.

Message: “ERROR: LNodeType <Object Name>: has no mandatory DO <Object Reference>: LN <object reference>.”

Meaning: Given mandatory DO is missing from given LN.

Message: “DO <Object Name>: SDO/DA <Object Reference>: DO <object reference>, inconclusive, please check manually”

Meaning: Some SDO (with given name) is unknown, possibly due to being in another name space. Please check if this DO is in another namespace and confirm the correctness of this SDO/DA not being available.

Message: “ERROR: DO <Object Name>: has no mandatory SDO/DA <Object Reference>: DA <object reference>.”

Meaning: Given mandatory SDO/DA is missing from given DO.

Message: “SDO <Object Name>: SDO/DA <Object Reference>: DA/Data attribute <object reference>, inconclusive, please check manually”

Meaning: Some SDO (with given name) is unknown, possibly due to being in another name space. Please check if the second SDO/DA is in another namespace and confirm the correctness of this first SDO not being available.

Message: “ERROR: DA <Object Name>: has no mandatory SDO/DA <Object Reference>: Data attribute not available.”

Meaning: Given mandatory Data attribute is missing from given SDO/DA.

Message: “Composite DA: none found in IEC61850, please check manually”

Meaning: A composite DA (structure or array) was found where the IEC61850 does not specify such a attribute. Please check if this data attribute is in another name space and the correctness of its availability.

Message: “DA <Object Name>: DA <Object Reference>: Data attribute <object reference>, inconclusive, please check manually”

Meaning: Some DA (with given name) is unknown, possibly due to being in another name space. Please check if this DA is in another namespace and confirm the correctness of the DA not being available.

Message: “ERROR: Composite DataAttribute <Object Reference>: DA <Object Name>: DA not available.”

Meaning: A composite data attribute (structure) has some missing DA..

Message: “ERROR: DA <Object Name>: Composite DataAttribute <Object Name>: DA not available.”

Meaning: A composite data attribute (structure such as a Point or a Vector) has some missing DA.

## **9.8. Messages in Mdl2**

Messages in Mdl2 are identical as in Mdl1 but then for conditional data attributes that must be available.

### **9.9. Messages in Mdl3**

Messages in Mdl2 are identical as in Mdl1 but then for conditional data attributes that cannot be available.

### **9.10. Messages in Mdl4**

Message: “ERROR: <Object class>: <Object Name> = <value>: <reason of failing>.”

Meaning: A name or object reference path is too long.

### **9.11. Messages in Mdl5**

Mdl5 does not generate error messages.

### **9.12. Messages in Mdl6**

Mdl5 does not generate error messages.

### **9.13. Messages in Mdl7**

Message: “ERROR: <Object Name>, <Object reference>: Type Mismatch: SCL <type>: IEC61850 <type>.”

Meaning: A type in the SCL (\*.icd/\*.cid) model is different from the expected type.

### **9.14. Messages in Mdl8**

Message: “DA <Object reference>: <type> Min(<value>) <= Value(<value>) <= Max(<value>). Value not in range, FAILING.”

Meaning: A value is out of range..

Message: “ERROR: DA <Object reference>: <type> (Enum): Min(<value>) <= Value(<value>) <= Max(<value>). Value not in range..”

Meaning: An enum value is out of range..

### **9.15. Messages in Mdl9**

Message: “ERROR: LN <Object reference>: At least one namespace missing. (InNs, cdcNs or dataNs)..”

Meaning: Expected at least one name space data attribute but none was found.

Message: “ERROR: LN <Object reference>: At least one namespace too much. (InNs, cdcNs or dataNs)..”

Meaning: Expected one name space data attribute but found more than one.

### **9.16. Messages in Mdl10**

Message: “ERROR: #<index> DA <Object Name>: <Object Ref1> and <Object Ref2> are out of order with each other.”

Meaning: Two DA attributes are in the wrong order..

### 9.17. Messages in Mdl11

Message: “ERROR: #<index> Name <Object Name>: <Object Ref1> and <Object Ref2> are out of order with each other..”

Meaning: Two data attributes are in the wrong order..

### 9.18. Messages in all Mdl tests

Message: “<Object class>: < Object Reference>, please check manually”

Example: “DO: MyLogicalDevice/LLN0\$EX\$MyDO, please check manually”

Meaning: Information about the object reference (of given object class) is not found in the IEC61850 reference data. The object reference might be an extension or enhancement. Correction: Check that the object reference is an extension or enhancement (see PICS/PIXIT information). If not, check that the object reference is not part of the IEC61850. Depending on the test, the test engineer has to validate this object reference and adjust the outcome of the test accordingly.

### 9.19. Messages available in all tests (Cnfx and Mdlx)

Message: ERROR: <Line number>: <Object class> <Object reference>: Missing <Object name>.”

Example: “ERROR: line (1846, 11): DA MyLogicalDevice/LLN0\$CO\$Mod\$Oper: Missing ‘Check’..”

Meaning: The data object ‘Check’ is missing in data object “MyLogicalDevice/LLN0\$CO\$Mod\$Oper” and as such, the test is FAILING.

Message: “ERROR: parent <Line number>: <Object class> <Object reference>: child <line number>: <Object name>: Not allowed.”

Example: “ERROR: line (1846, 11): DA MyLogicalDevice/LLN0\$CO\$Mod\$Oper: child (1890, 11): ‘Check’: Not allowed.”

Meaning: The data object ‘Check’ is included in data object “MyLogicalDevice/LLN0\$CO\$Mod\$Oper” while not allowed and as such, the test is FAILING

Message: “Start: <test>”.

Example: “Mdl5: Start: Mdl5”

Meaning: Test Mdl5 has started, any log message from within the test is preceded with the <test> name, including this one.

Message: “Test description: <some text>”

Example: “Mdl5: Test description: Verify data model mapping according to...”

Meaning: This is what test Mdl5 is specified to do (according to the UCA).

Message: “Start Conformance test.”



Example: “*Start Conformance test.*”

Meaning: Test script initialisation is done, the actual test is starting now, any messages after this message do imply the actual test and its results.

Message: “Finished Conformance test, result: <result>.”

Example: “*Finished Conformance test, result: PASSED.*”

Meaning: The actual test is done and the final result is known. Some de-initialisation might follow to clean up test results.

Message: “Done: <test>”.

Example: “*Mdl5: Done: Mdl5*”

Meaning: Test Mdl5 has finished; this is the last log message from within the test, still preceded with the <test> name.

Message: “No valid SCL data from file available (test inconclusive)”

Example: “*No valid SCL data from file available (test inconclusive)*”

Meaning: You did not load any SCL file (\*.icd etc) but information from it is needed for the test you selected to run.

Correction: Load the SCL file (\*.icd etc) belonging to the DUT that you are testing to correct this.

Message: “No valid DUT data from model available (test inconclusive)”

Example: “*No valid DUT data from model available (test inconclusive)*”

Meaning: You did not build a data model for the DUT but information from it is needed for the test you selected to run.

Correction: Connect to a DUT and build a data model.

Message: “ERROR: <Some text>: Unforeseen problem.”

Example: “ERROR: divide by zero error: Unforeseen problem.”

Meaning: Due to some corrupt data, programming error or other reason not covered in testing the program has met a failing point from which it cannot recover. Due to that the test cannot continue and must fail. This message might imply a bug in the program but could as well be a bug or non-compliance condition in the DUT. Please report this message to GridClone. Any details on how the message was triggered, what SCL files and DUT are used is immensely helpful by solving the problem.

## 10. Additional Topics

### 10.1. *File name conventions*

Any \*.txt file is a test result file and has the same content as the log frame.

Any \*.log file is an error log file for errors related to the program and not to the test.

**Mdl1-2013-01-07\_09.28.43-CS.txt** is the test result of test case *Mdl1* generated by the *SCL Checker* on the 7<sup>th</sup> of January 2013 at 09:28 local time.

**2013-01-07\_09.28.43-SimFlexError.log** is an error log for the same run. This file is located in the “Application base directory”.

### 10.2. *Used abbreviations*

DUT            Device Under Test.

IEC            The International Electrotechnical Commission, managing international standards. See: <http://www.iec.ch/>

IEC 61850    The IEC 61850:2003 standard.

IED            Intelligent Electronic Device

UCA           The UCA® International Users Group. See: <http://www.ucaiug.org>

### 10.3. *Referenced documents*

IEC 61850    IEC 61850:2003

MMS           ISO 9506:2003 (as used by IEC 61850-8)  
Getting started.pdf

### 10.4. *Referenced tools*

WinPcap      For capturing network traffic, see: <http://www.winpcap.org/>

WireShark    For analysing network traffic, see: <http://www.wireshark.org/>

### 10.5. *Version History*

Version	Date	By	Subject
1.0	January 1, 2013	EM	Initial version
1.1	January 22, 2013	EM	Extending on several subjects.
1.2	January 26, 2013	EM	General rework on entire document